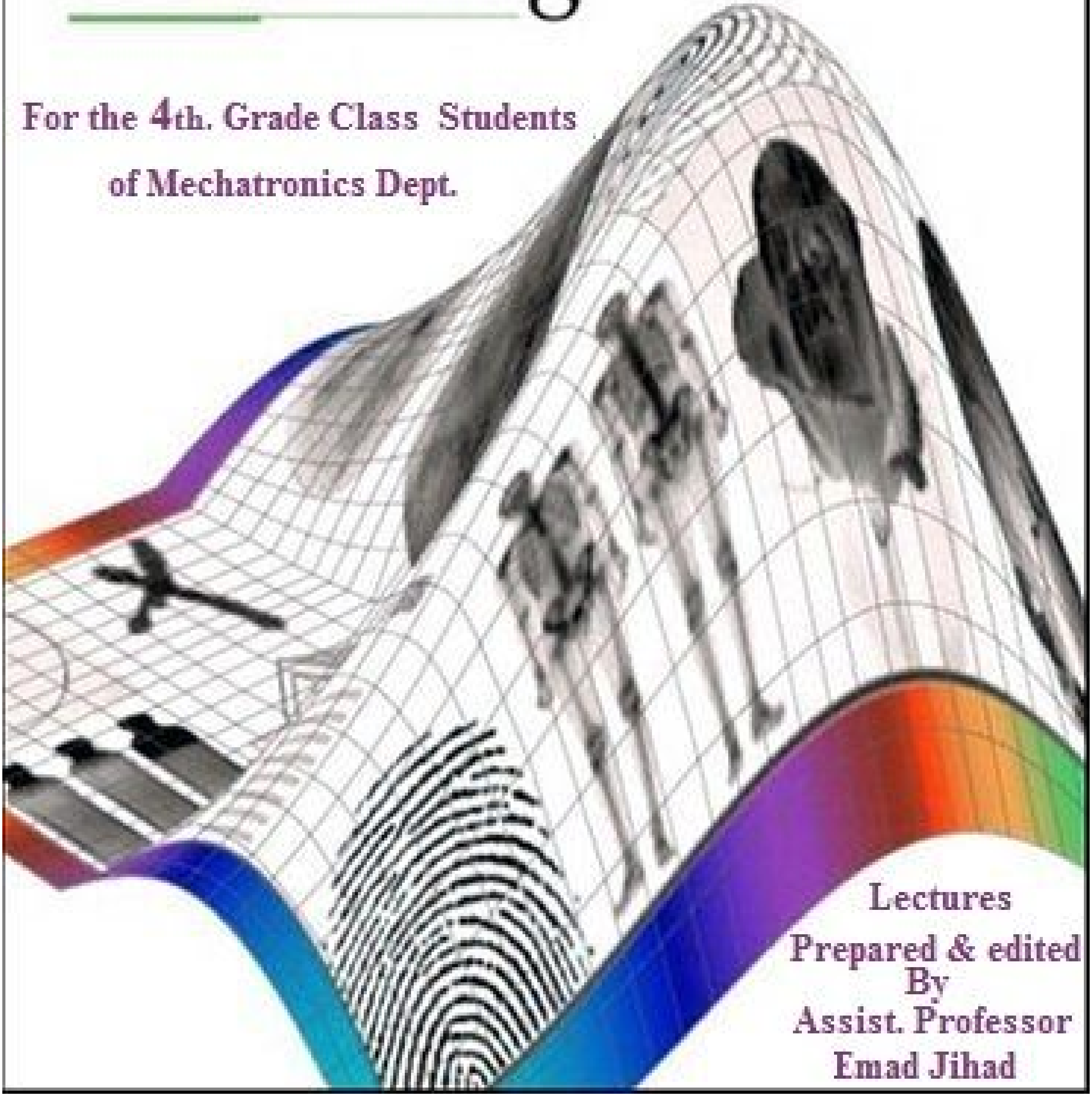
Digital Image Processing USING MATLAB

For the 4th. Grade Class Students of Mechatronics Dept.



Lectures
Prepared & edited By
Assist. Professor
Emad Jihad

1- Digital Image Processing (*the basics introduction*)

Digital image processing is the use of computer algorithms to perform image processing on digital images. Since images are defined over two dimensions (perhaps more) digital image processing may be modeled in the form of multidimensional systems.

image processing is any form of signal processing for which the input is an image, such as a photograph or video frame; the output of image processing may be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it.

Why do we need image processing?

1- Improvement of pictorial information for human perception (Private and Public).

2- For Autonomous machine application. These may include:

a- Industrial machine vision for Product assembly and inspection.

b- Automated Target detection and tracking.

c- Finger print recognition.

d- Arial and Satellite imagery (Weather prediction, Crop assessment, Security and Military…etc.).

As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortion during processing.

Images come in Four **types**:

1- Black and White, or Binary
2- Gray level (gray scale)
3- Color (RGB) images.
4- Indexed (Color) images.

Any image of any type is consisting of small points called Picture Elements (**Pixels**). Images are represented in 2 dim arrays (matrices) as sets of these pixels.

In **Black and white** (B/W) images, each pixel has either the value of 0 or 1 (Binary), in which the 0's represent Black pixels, while the 1's are the Whites. (Fig.1-1).

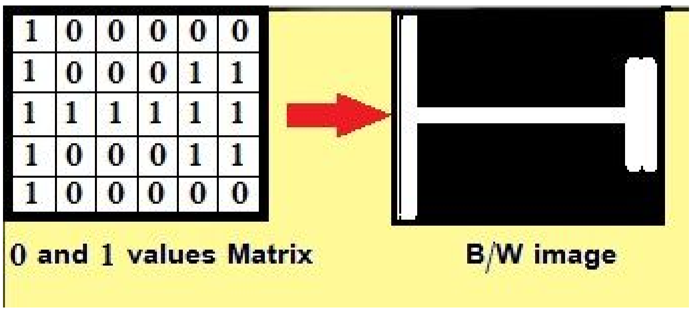
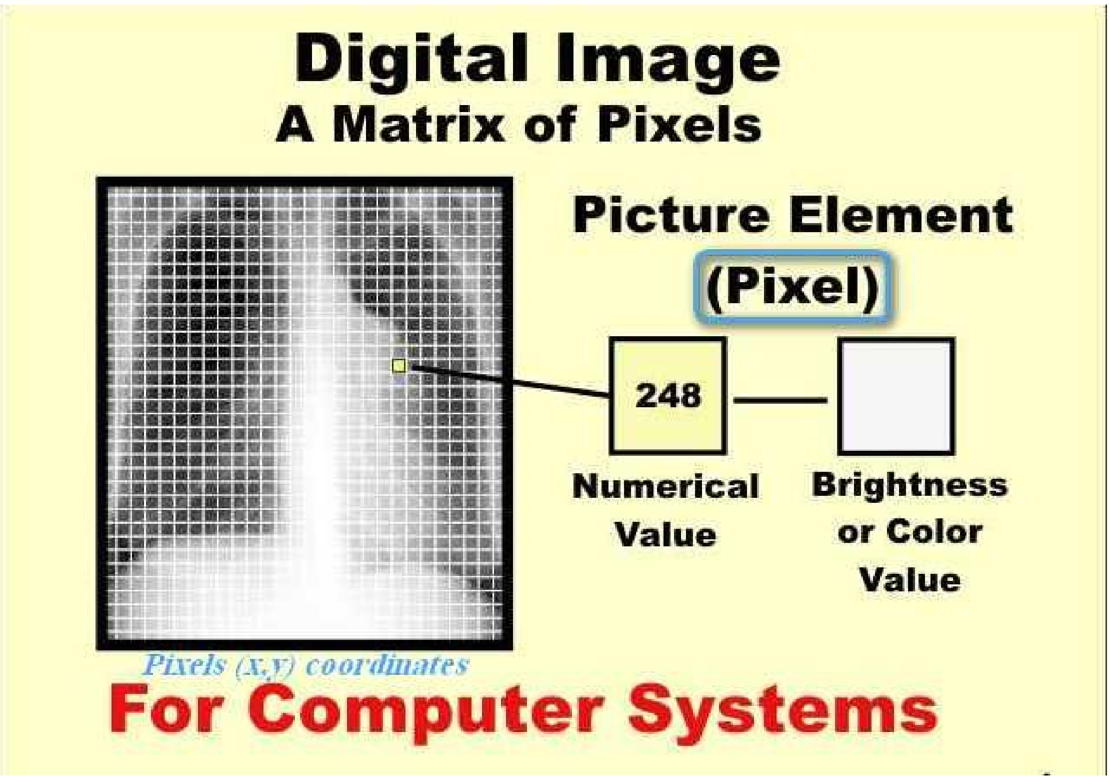


Fig. (1-1)

An Array of **Gray scale** image needs to include numbers represent the ***Pixels intensity*** value and the ***Gray level***.
Gray level will be (0) for Black level and (255) for the White level, while other in Between levels should be (> 0 and < 255) according to the pixels Gray levels.
(see Fig.1-2)



Since gray level images are taking less storage space, thus most image processing application consider these images as excellent according to the speed of accessing and processing (Fig. (1-3)).

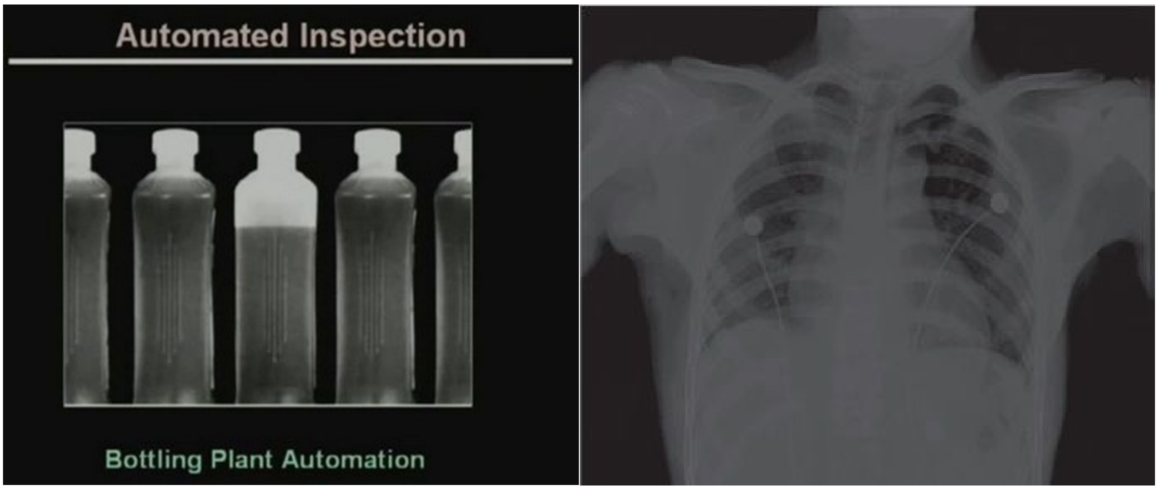


Fig. (1-3)

For **Color images**, we need to define (3) 2-dim. Arrays to hold the values of the three main colors **RGB** (Red, Green, and Blue) for each pixel. (See Fig.1-4).

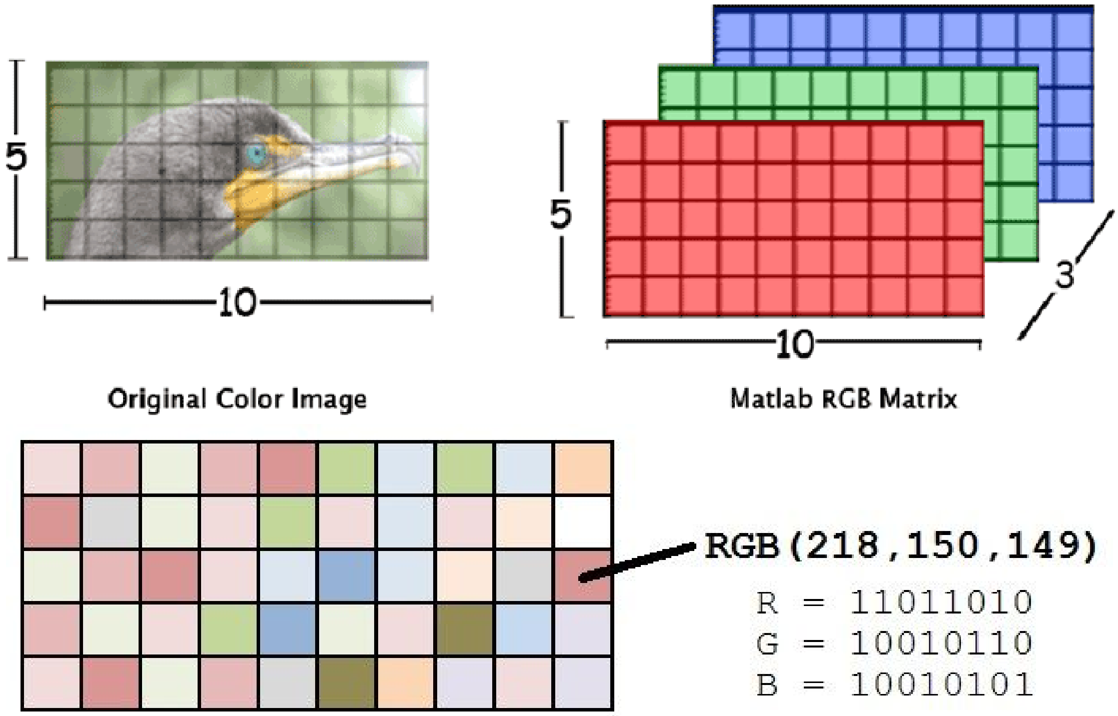


Fig. (1-4)

Hence,

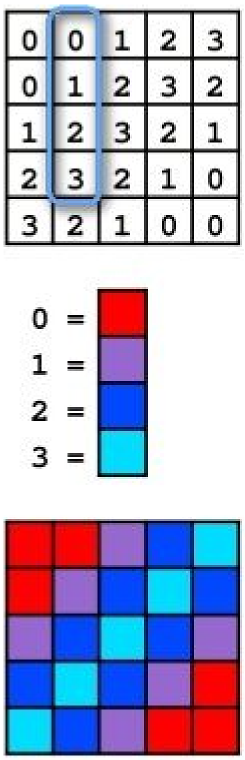
- $p = (r,c)$ is the pixel location indexed by row, r, and column, c.
- $I(p) = I(r,c)$ is the value of the pixel at location p.
- If $I(p)$ is a single number then I is monochrome.
- If $I(p)$ is a vector (ordered list of numbers) then I has multiple bands (*e.g.*, a color image).

With respect to the *x*- and *y*-coordinates, and also in amplitude. Converting such an image to digital form requires that the coordinates, as well as the amplitude, be digitized.
Digitizing the coordinate values is called ***sampling***; digitizing the amplitude values is called ***quantization***. Thus, when x, y, and the amplitude values of f are all finite, discrete quantities, we call the image a ***digital image***.

Assume that an image $f(x,y)$ is sampled so that the resulting image has M rows and N columns. We say that the image is of *size* $M * N$. The values of the coordinates are discrete quantities.
The image origin is defined to be at $(x,y) = (0,0)$. Note that x ranges from 0 to M-1 and y from 0 to N-1 in integer increments.

However, Color images are more complicated in processing and need more storage space than B/W and Gray level images.

At last, we have the indexed color images.



Indexed color is a technique to manage digital images' colors in a limited fashion, in order to save computer memory and file storage, while speeding up display refresh and file transfers. It is a form of vector quantization compression.

When an image is encoded in this way, color information is not directly carried by the image pixel data, but is stored in a separate piece of data called a **palette**: an array of color elements, in which every element, a color, is indexed by its position within the array. The image pixels do not contain the full specification of its color, but only its index in the *palette*. This technique is sometimes referred as **pseudocolor** or **indirect color**, as colors are addressed indirectly.

Fig. (1-5) represents a comparison between Color and indexed color images.

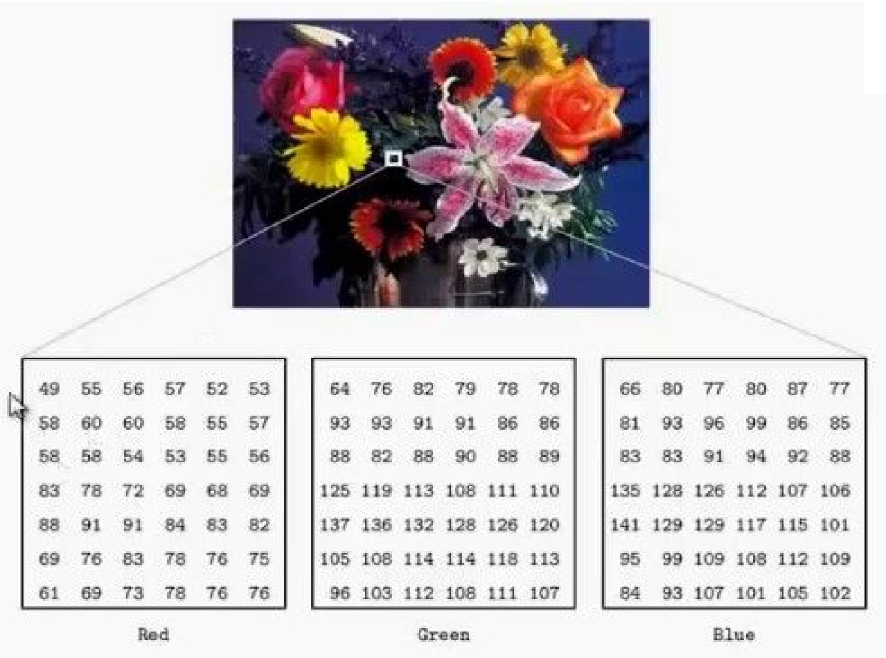
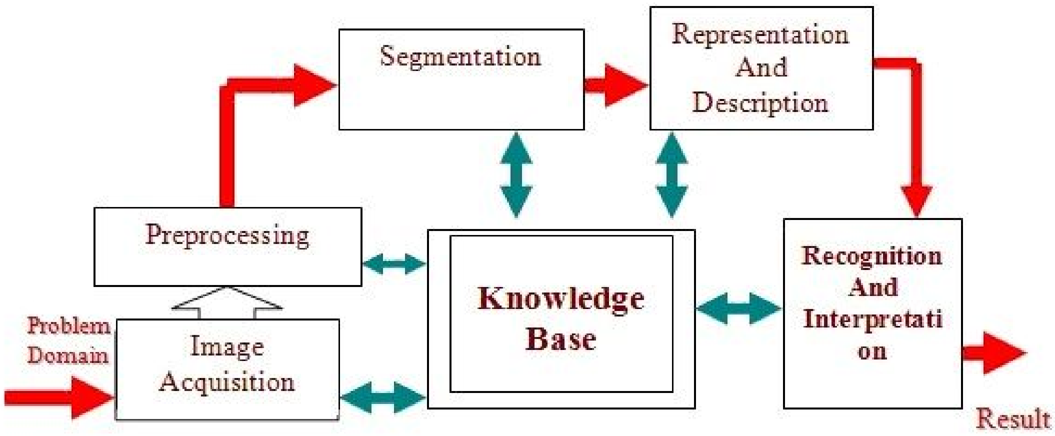


Fig. (1-5)

The complete Steps in Digital Image Processing:

The following diagram is showing the complete image processing steps for ideal applications.



columns. The other difference is that the origin of the coordinate system is at $(r,c) = (1,1)$; thus, r ranges from 1 to M, and c from 1 to N, in integer increments. (See Fig. 1-7).

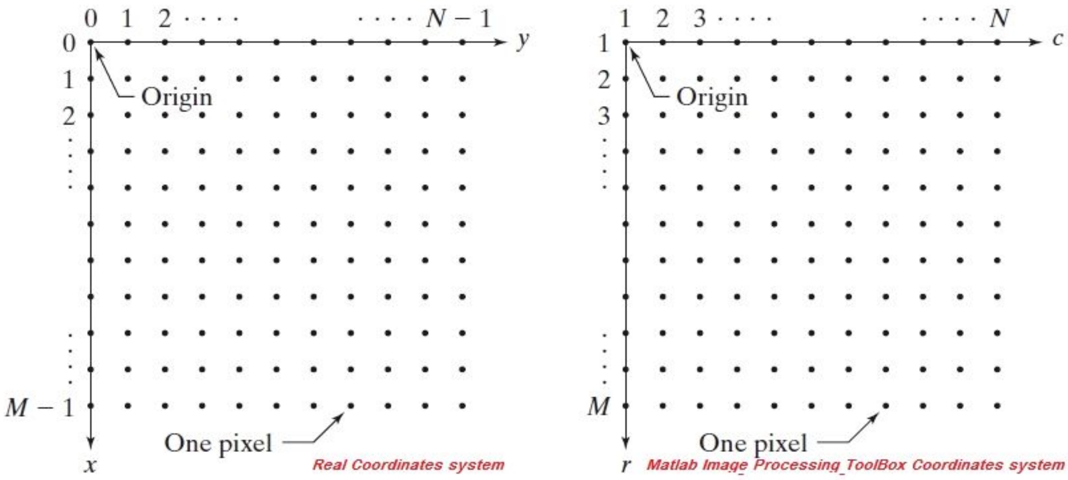


Fig. (1-7).

Thus, the coordinate system in Fig. (1-7) representation for a digitized image (both) will be:

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix}, \quad \texttt{f} = \begin{bmatrix} \texttt{f(1,1)} & \texttt{f(1,2)} & \cdots & \texttt{f(1,N)} \\ \texttt{f(2,1)} & \texttt{f(2,2)} & \cdots & \texttt{f(2,N)} \\ \vdots & \vdots & & \vdots \\ \texttt{f(M,1)} & \texttt{f(M,2)} & \cdots & \texttt{f(M,N)} \end{bmatrix}$$

Where f (1, 1) = f(0, 0)

Image Resolution

Each image has a resolution that depends on the number of pixels in both rows and columns (or width and height). This resolution depends on the device used to capture the image such as digital cameras, scanners or computer application that generating it.

Example (1-1): An image that is 2048 pixels in width and 1536 in height has a total image size of:
2048 X 1536 = 3,145,728 pixels or 3.1 MP (Mega Pixels).

Spatial Resolution

Unlike image pixel resolution which count pixels number (some times per inch), the spatial resolution represents empty lines between image pixels rows. The closest lines the better image clarity given.

Image File Structure

Images are stored and retrieved in different pixel formats along with file structure seen below (Fig. (1-8)).

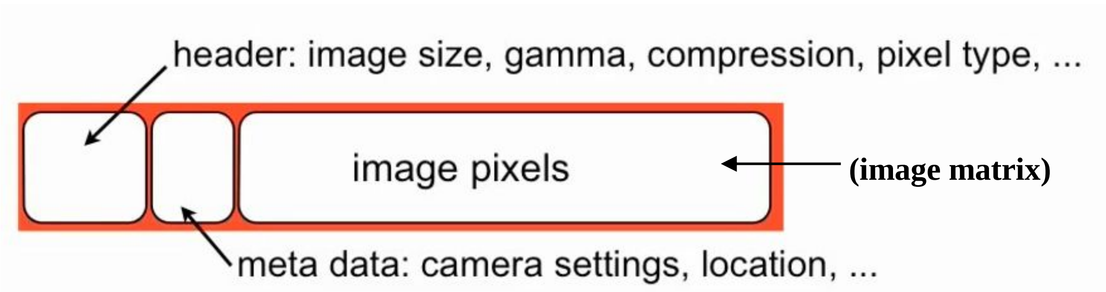


Fig. (1-8)

Components of header part:

Image size is the **resolution** in pixels (KP, MP, and GP) as mentioned before.
It is Not the same file size measured in (KB, MB, and may be GB). This file size is important for the purpose of storage space required.

Example (1-2): Compute an image file size of a format pixels size = 3 Bytes and has a resolution of 2048 X 1536.
Solution:

2048 X 1536 = 3,145,728 pixels, and since each pixel requires 3 Bytes, then:
3,145,728 X 3 = 9,437,184 Bytes or 9 MB.
(Note: this is a theoretical computation, in practical it is less than this as will be seen).

Gamma is the Brightness or intensity of an image **or image Luma** (Luminance) (color images), it is set by the device used (such as digital camera) or can be manipulated by the user. However, this value ranges between 0.0 and 1.0, and given by the relation: $L^{\frac{1}{\gamma}}$ Where γ is the Gamma value.

<u>Example (1-3</u>): A camera has a gamma of 0.45, which of these cases are Brighter:
Case pixel A = 0.3 or case pixel B = 0.6?
Solution:
By using the Luma formula: $L^{\frac{1}{\gamma}}$
We get:

Pixel A: $(0.3)^{1/0.45} = 0.0689$

Pixel B: $(0.6)^{1/0.45} = 0.3214$
Thus Pixel B is 3 times Brighter than Pixel A.

Compression or image compression are the techniques used to reduce image size in order to get as small storage size as possible and make transmission faster.
There are two main techniques to compress images:

1- Lossless: Lossless compression is preferred for archival purposes and often for medical imaging, technical drawings, clip art, or comics. Image formats using this method are PCX, BMP, TIFF, and PNG….etc.
2- Lossy: Lossy methods are especially suitable for natural images such as photographs in applications where minor loss of fidelity is acceptable. JPG is the most used Lossy compressed image format.

Finally**, Pixel type** refers to number of bits required for a single pixel depending on image format and type. A **B/W** image pixel has only One bit, therefore it is called sometimes Binary image (either 0 for Black or 1 for White pixel color).

A **Gray** scale (level) image pixel has One Byte (8bits) for each pixel, that gives the gray level (0~255) or (0 ~ 11111111 in binary) that gives the intensities between Black and White. In Matlab, such images will have what's called unit8 class.

An **RGB** color image pixels, have 3Bytes or 24bits for each pixel, 1Byte for Red level, 1 Byte for Green level and another Byte for Blue level. It's also unit8 class.

Components of Meta data:

These are additional data considered as references of the image such as the name of digital Camera used, setting of that camera in the time of the image taken, also the location of the image (if GPS or location option was set by the user)….etc.